```
                .DO     External
                .LSTON
                .Page
;>>>>>>>>>>>>>>>>>>>>>>>>>
;>
;>      Module: Srvo1.bl
;>
;>      This module contains all the specifications and source code for
;>      controlling the Widget Servo Board ( i.e., communication
;>      protocol, seek and head positioning, spare table lookup, etc. )
;>
;>      PROCEDURE Seek( Cylinder : WORD { !!rC }
;>                      Head : BYTE { !rE }
;>                      Sector : BYTE { !rF }
;>                    )
;>      PROCEDURE Wait_xMs( x : BYTE { !r1 }
;>      FUNCTION PosHeads( Wait : BOOLEAN { !r8/bit 6 }
;>                         Parent : BYTE { !rB }
;>                         Cylinder : WORD { !!rC }
;>                         Head : BYTE { !rE }
;>                         Sector : BYTE { !rF }
;>                       ) : BOOLEAN
;>      PROCEDURE SelectHead( Head : BYTE { !rE } )
;>      FUNCTION CalcMagDir( Cylinder : WORD { !!rC } ) :
;>                         BOOLEAN
;>                         Magnitude : WORD { !r5 }
;>                         Direction : BYTE { !r9 }
;>      FUNCTION FindDistance( Cylinder : WORD { !!rC } ) : BOOLEAN
;>      FUNCTION ServoOk : BOOLEAN
;>                         RecalMagDir : BYTE { !r0 }
;>      PROCEDURE SelectHead( Head : BYTE { !rE }
;>      PROCEDURE Auto_Offset
;>      PROCEDURE Chk_Offset
;>
;>>>>>>>>>>>>>>>>>>>>>>>>>
                .LSTOFF
                .FIN
                .DO     External
                .LSTON
                .Page
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
;>
;>      Procedure: Seek
;>
;>      This procedure accepts cylinder, head, and sector values
;>      and then calls position heads.
;>
;>      Inputs:
;>              Cylinder: WORD { !!rC }
;>              Head    : BYTE { !rE }
;>              Sector  : BYTE { !rF }
;>
;>      Outputs: { none }
;>
;>      Global Varibles Changed:
;>              Cylinder, Head, Sector
;>
;>      Algorithm:
;>
;>      BEGIN
;>       LastSeekAddress := CurrentSeekAddress
;>       DiskStat.Parked := False
;>       IF DiskStat.SeekComplete
```

```
;>        THEN
;>             i := 4
;>             WHILE ( i >       0 ) AND NOT( PositionHeads( Wait,
;>                                    Dmt_Seek, Cylinder, Head, Sector ) DO
;>
;>              IF NOT( Recovery ) THEN Abort
;>              i := i -1
;>
;>             IF ( i = 0 )
;>              THEN Abort
;>          ELSE Wait up to 1 sec for ServoReady, if timeout Abort
;>         LastCylinder := GlobalCylinder
;>         LastHead     := GlobalHead
;>         LastSector   := GlobalSector
;>         GlobalCylinder := Cylinder
;>         GlobalHead     := Head
;>         GlobalSector   := Sector
;>         SectorCount := SectorCount + 1
;>         DiskStat.SeekComplete := True
;>         DiskStat.OnTrack := True
;>       END
;>
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
                .LSTOFF
                .FIN
                .DO     Internal
                .LSTON
                .Page
                .FIN

Seek:
                Call    Ext_Push

                Ld      Lst_HiCyl,Cylinder ;save last seek address
                Ld      Lst_LoCyl,Cylinder+1
                Ld      Lst_Head,Head
                Ld      Lst_Sector,Sector
                And     DiskStat,#$FF-Parked

                Tm      DiskStat,#SeekComplete ;check for head/arm motion
                Jr      Z,Seek_Wait

Seek_ReTry:     Ld      !r1,#4
                And     DiskStat,#$FF-Long_Seek

Seek_Lp:         Ld     !r8,#Wait
                Call    PosHeads
                Jr      Nz,Seek_End

                Tm      Excpt_Stat,#Recovery
                Jr      Z,Seek_Abt
```